What is claimed is:

1. A method for programmatically enforcing referential integrity constraints among associations between class instances, comprising steps of:

evaluating a request to set an association end to reflect an association from an instance of a first class to an instance of a second class;

setting the requested association end; and

programmatically modifying an inverse association end of the association to reflect an inverse association from the instance of the second class to the instance of the first class.

2. The method according to Claim 1, wherein the evaluating step further comprises determining whether the association end has a single multiplicity or a many multiplicity.

3. The method according to Claim 2, wherein the setting and programmatically modifying steps for a particular association end that has a single multiplicity further comprise steps of:

disconnecting the inverse association end from an existing instance, if any;

performing the programmatically modifying step after performing the disconnecting step; and

performing the setting step after performing the disconnecting step.

4. The method according to Claim 2, wherein the setting and programmatically modifying steps for a particular association end that has a many multiplicity further comprise steps of:

performing the setting step;

disconnecting the inverse association end from an existing instance, if any, after performing the setting step; and

performing the programmatically modifying step after performing the setting step.

5. The method according to Claim 1, further comprising steps of:

determining whether the association end or the inverse association end is a primary end of the association; and

serializing only the primary end of the association during a serialization operation.

6. The method according to Claim 1, wherein the method is provided as link helper objects.

7. A computer program product for programmatically enforcing referential integrity constraints among associations between class instances, wherein the computer program product is embodied on one or more computer readable media and comprises:

computer-readable program code means for evaluating a request to set an association end to reflect an association from an instance of a first class to an instance of a second class;

computer-readable program code means for setting the requested association end; and

computer-readable program code means for programmatically modifying an inverse association end of the association to reflect an inverse association from the instance of the second class to the instance of the first class.

8. The computer program product according to Claim 7, wherein the computer-readable program code means for evaluating further comprises computer-readable program code means for determining whether the association end has a single multiplicity or a many multiplicity.

9. The computer program product according to Claim 8, wherein the computer-readable program code means for setting and the computer-readable program code means for programmatically modifying for a particular association end that has a single multiplicity further comprise:

computer-readable program code means for disconnecting the inverse association end from an existing instance, if any;

computer-readable program code means for operating the computer-readable program code means for programmatically modifying after operating the computer-readable program code means for disconnecting; and

computer-readable program code means for operating the computer-readable program code means for setting after operating the computer-readable program code means for disconnecting; and

wherein the computer-readable program code means for setting and the computer-readable program code means for programmatically modifying for a particular association end that has a many multiplicity further comprise:

computer-readable program code means for performing the setting;

computer-readable program code means for disconnecting the inverse association end from an existing instance, if any, after operation of the computer-readable program code means for setting; and

computer-readable program code means for operating the computer-readable program code means for programmatically modifying after operating the computer-readable program code means for setting.

11. The computer program product according to Claim 7, further comprising:

computer-readable program code means for determining whether the association end or the inverse association end is a primary end of the association; and

computer-readable program code means for serializing only the primary end of the association during a serialization operation.

12. A system for programmatically enforcing referential integrity constraints among associations between class instances, comprising:

means for evaluating a request to set an association end to reflect an association from an instance of a first class to an instance of a second class;

means for setting the requested association end; and

means for programmatically modifying an inverse association end of the association to reflect an inverse association from the instance of the second class to the instance of the first class.

13. The system according to Claim 12, wherein the means for evaluating further comprises means for determining whether the association end has a single multiplicity or a many

multiplicity.

14. The system according to Claim 13, wherein the means for setting and the means for programmatically modifying for a particular association end that has a single multiplicity further comprise:

means for disconnecting the inverse association end from an existing instance, if any;

means for operating the means for programmatically modifying after operating the means for disconnecting; and

means for operating the means for setting after operating the means for disconnecting; and

wherein the means for setting and the means for programmatically modifying for a particular association end that has a many multiplicity further comprise:

means for performing the setting;

means for disconnecting the inverse association end from an existing instance, if any, after operation of the means for setting; and

means for operating the means for programmatically modifying after operating the means for setting.

15. The system according to Claim 11, further comprising:

means for determining whether the association end or the inverse association end is a primary end of the association; and

means for serializing only the primary end of the association during a serialization operation.